Towards An Early Design Space Exploration Tool Set for Spin Transfer Torque RAM (STT-RAM) Design

Philip Asare
pka6qz@virginia.edu

Ben Melton
bwm3hk@virginia.edu

ABSTRACT

Spin transfer torque random access memory (STT-RAM) is a promising new memory technology which is a candidate for universal memory. However, because of the infancy of this technology, there is a limited set of models and tools available for general cross-layer analysis which is essential for early design space exploration. In this paper, we show how two existing cross-layer analysis tools for SRAM can be extend to provide insights on how decisions in the layers of abstraction affect array-level energy and delay characteristics. This approach provides circuit designers with the much needed cross-layer perspective for STT-RAM and the ability to compare both memory technologies (SRAM and STT-RAM) within the same set of tools. Our energy and delay results based on varying parameters at different layers are consistent with results found in the literature.

1. INTRODUCTION

Spin transfer torque random access memory (STT-RAM) is an emerging memory technology that promises to deliver the benefits of current non-volatile memories (speed, high density) with the added benefit of being non-volatile and offering no leakage power from the storage element [11]. However, before these benefits can be reaped, memory designers (both architects and circuit designers) must be able make informed design decisions to meet specific application needs while minimizing unwanted effects. Memory design, like any integrated circuit design, requires a cross-layer approach. The layers of abstraction provide the ability to tame the complexity of designing such systems; however, they now present a challenge as design decisions made in one layer impact the decisions that can be made in other layers. Designers that operate in one layer must, therefore, be able to understand the impact that their decisions have on other designers and vice versa. Design tools must be able to interact in ways that allow designers to see effects across the layers of abstraction.

Two tools, the Technology-Agnostic Simulation Environment (TASE)[5] and the Virtual Prototyper (ViPro)[4], have been developed in the static random access memory (SRAM) domain. TASE works at the interface between process technologies and the circuit and device levels, providing insights to the designer on how different circuit and device decisions play out across different process technologies. It does this by allowing the designer to specify a simulation template which is used across different technologies, eliminating the need to create new simulation specifications for each technology and each change in the circuit. TASE works with Cadence and uses Ocean scripting, two tools that circuit designers are already familiar with. ViPro operates at the interface between the circuit and architecture levels, allowing circuit designers to specify virtual prototypes of SRAM memory arrays with varying levels of detail and to see how various architecture and circuit level decisions affect the energy-delay characteristics of the whole array. ViPro is currently implemented in MATLAB, though its models can be implemented in other object-orient programming tools. Both tools can be combined (ViPro can operate on TASE output) to give the circuit designer insights across all the layers.

Our long-term goal is to extend these tools to support STT-RAM designs, allowing circuit designers to do the same early design space exploration, mentioned above, that these tools allow for SRAM. The benefit of this approach (extending the SRAM tools) is that SRAM and STT-RAM can now be compared within the same tools providing circuit designers and architects insights on where each technology is beneficial in the memory hierarchy. This work is targeted at designers who work at the interface between architecture and the circuit level. Hence, though it will not give detailed insights into how the memory might perform when implemented in a full system, it provides the information needed to evaluate such performance more accurately and allows circuit designers to make more informed optimization decisions. In this paper, we show how the SRAM tools can be extended with models for STT-RAM and present preliminary but promising results for a number of array configurations with parameters varying across the process, circuit, and architecture levels.

2. RELATED WORK

Much of the current work in STT-RAM modeling has been either been process-specific and/or mainly focused on the architecture level, where the memory is investigated in simulated implementations to understand its impact on processor performance. Also, many of the works focus on modeling STT-RAM arrays for specific scenarios, rather than providing general tools for exploring the STT-RAM design space. Smullen *et al.* looked at improving delay performance of STT-RAM to be used in caches by reducing the retention time and hence relaxing the non-volatility of the memory [8]. These investigations were carried out in CACTI, an architecture-level modeling tool for memory performance using the 32nm process technology from the ITRS roadmap. Nigam *et al.* investigated reducing the write energy of STT-RAM by proposing device level changes (changing the storage element material) and using an inverting code scheme

on the architecture level [7]. The process technology used was not mentioned, however there was no mention of tests across different process technologies. Smullen *et al.*, in other work, proposed a modeling framework for memory systems researchers, again concentrating on the architecture level by using a first-order model of the magnetic tunnel junction (MTJ), the STT-RAM storage element [9]. Chatterjee *et al.* looked at circuit-level methods (varying access transistor width and array supply voltage) for minimizing energy dissipation while maintaining acceptable memory performance for an L2 cache [2]. Their work was evaluated using the 180nm TSMC technology provided by MOSIS[1]. Our work adds to all these efforts by providing the necessary information needed from the lower levels of abstraction in order to get more accurate estimates of architectural metrics of interest. TASE and ViPro, mentioned previously, are tools that can be combined to provide a cross-layer perspective for memory design; however, these tools were designed specifically for SRAM. Our main contribution is extending them to provide a similar cross-layer perspective for STT-RAM design.

3. STT-RAM OVERVIEW

The storage element for STT-RAM is the magnetic tunneling junction (MTJ), a magnetic structure with three layers: the fixed layer, oxide layer, and free layer. The fixed and free layers are ferromagnetic layers with fixed and variable magnetic orientation respectively. When electric current is passed through the MTJ, the current's electron spin is polarized by the fixed layer, and must tunnel through the oxide layer. After passing through the oxide layer, the electron spin induces a magnetic torque on the free layer, either attempting to reverse or reinforce its magnetic orientation. An electron current flowing from the fixed layer to the free layer is polarized in the direction of the fixed layer, and induces a torque on the free layer that is parallel to the fixed layer. An electron current flowing from the free layer to the fixed layer is polarized and reflected by the fixed layer back into the free layer. This reflection reverses the spin of the electrons, and induces a torque on the free layer that is anti-parallel to the fixed layer. The magnetic state of the free layer relative to the fixed layer, either parallel or anti-parallel, determines the resistance of the device to an applied voltage. This binary low or high resistance provides the two states for digital memory. Combined with an access transistor, this circuit creates a bit cell, which is the principal storage component of an STT-RAM memory structure.

To access this device, the MTJ side of the bit cell is connected to the bit line, and the transistor side is connected to the source line. A word line runs perpendicular to the bit lines to enable the access transistor of the bit cells. The bit line and source line are connected to four write control transistors, one at the top and bottom of both lines. These transistors control current direction for writes to the bit cells. The bit line is connected to a sense amplifier. Since the MTJ is a current device, a current sense amplifier is used to detect the difference in current between the bit cell and a reference. In our investigations, we use power and delay metrics from a clamped-bit-line current sense amplifier[1]. The general structure described here is shown in Figure 1.

[1]http://www.mosis.com

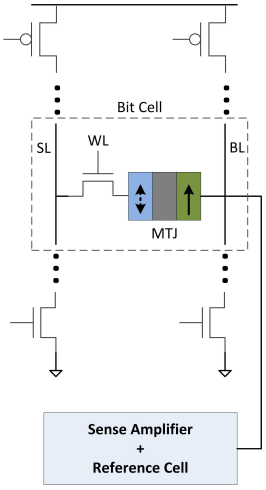

Figure 1: STT-RAM bit column. Write transistors at the top and bottom are used for the whole column. BL = bit line, WL = word line, SL = source line, single arrow = MTJ fixed layer, double arrow = MTJ free layer

4. STT-RAM MODELING FOR DESIGN SPACE EXPLORATION TOOLS

4.1 Technology-Agnostic Simulations

To characterize the voltage-current relationship of the bit cell, we wrote an STT-RAM template for TASE. A TASE 'test' consists of a Spectre netlist, the Ocean script template to run the netlist, and a MATLAB script template to plot the results (the MATLAB output was suppressed for our tests since ViPro only requires TASE's text output). The Ocean script template is an Ocean script file that contains variable parameters which TASE can substitute at runtime. This allows the user to write circuit simulations that are independent of technology, voltage, or other generally fixed parameters. Several of these tests can be combined into simulation templates, which define which tests to run, and what parameters to pass to each test. This way, an entire characterization of a bit cell can be performed simply by writing test simulations for every desired data set, and running a template that calls each test with the desired technology, voltage, sizing, and other parameters. Our test applies a range of write voltages (0.5V to 1.5V) and a range of read voltages (0 to 20mV) to an STT-RAM bit cell netlist, which uses the MTJ model developed by Nigam *et al.*[6]. TASE records the current through the bit cell for each voltage for each read and write scenario. There are four write scenarios: flipping the cell (0 to 1 or 1 to 0) and writing the same value to the cell (1 to 1 or 0 to 0). The test repeats this simulation for different access transistor widths (one to four times the minimum width), and finally for three different technology nodes, 22nm PTM, 45nm PTM, and 90nm PTM which are included with TASE. The resulting data is formatted in a text file as a table associating each width and voltage tuple to the measured currents for each scenario. This table is then passed to ViPro to provide data for energy and delay calculations of the bit cell.

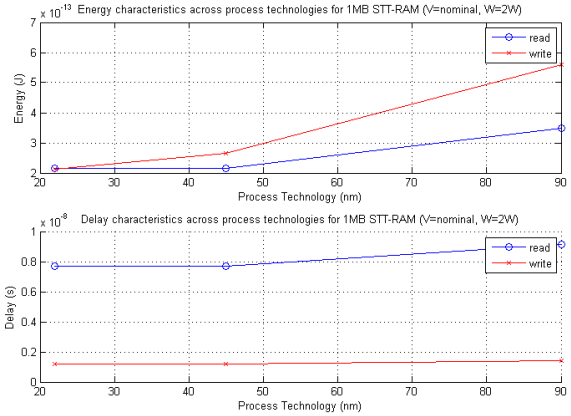

Figure 2: Process technology effect on energy and delay

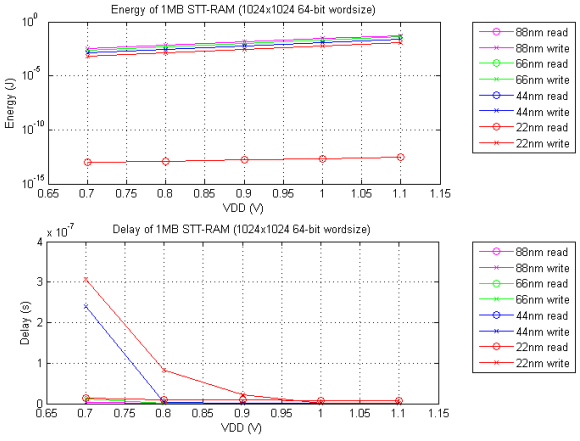

Figure 3: Circuit-level parameter effects on energy and delay (voltage perspective)

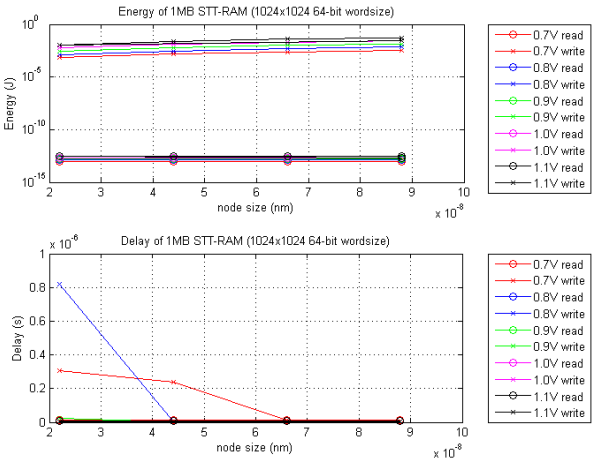

Figure 4: Circuit-level parameter effects on energy and delay (access transistor width perspective)

4.2 Virtual Prototyping

ViPro uses that data from TASE, combined with other information like sense amp delay and read voltage for reads, and total delay and supply voltage for writes, to calculate the energy and delay of the bit cell. The energy and delay values for the sense amplifier were generated in HSPICE for the same read scenarios simulated in TASE and hard-coded into the sense amplifier model. The energy and delay values for the decoders were produced by developing an analytical model based on information from work by Turi and Delgado-Frias in optimization of 4-to-16 memory decoders [10]. The model takes supply voltage and number of rows or word size as an inputs and produce energy and delay as a output. The cell dimensions, essential for determining bit line, source line, and word line lengths were based on the model used by Chatterjee *et al.* [2]. In order to calculate the write time of the bit cell, we modified the model used in [9] and [3] to provide the write time as a function of the average write current. Since read and write operations happen in step-by-step fashion (*i.e.* one component is turned on before the next to ensure correct operation), the delay for each operation was found as the sum of the delays of all components involved. In the read operation, the main components that contribute to delay are the sense amplifier (which has to be pre-charged), the word line, and the decoders. In the write operation, the main components that contribute to delay are the decoders, the word line, the bit or source line (depending on the value being written), and the bit cell. For the energy, we sum up the energy of all components that are on (or leaking) during the operation.

5. EXPERIMENTS AND RESULTS

Our experiments were targeted at showing how different changes at different layers of abstraction affected the energy and delay characteristics of the whole array. We kept the periphery circuits the same throughout the experiments. At the process level, the main knob (independent variable) was the process technology. We investigated three different process technologies: 22nm PTM, 45nm PTM, and 90nm PTM. The results for a 1MB array (1204x1024, 32-bit word size) are shown in Figure 2. Each process technology is at nominal voltage and the access transistor width is twice the minimum width.

At the circuit level, the knobs were the supply voltage and the access transistor width. The results for varying these parameters for a 1MB array (1024x1024, 64-bit word size) are shown in Figures 3 (voltage perspective) and 4 (access transistor width perspective). This experiment was run for the 22nm PTM.

At the array level, the knobs were number of rows (and hence columns) and the word size. The results for varying these parameters for a 1MB array in the 22nm PTM at 0.9V are shown in Figure 5.

6. DISCUSSION

Our tests sweep many of the parameters used as knobs for optimizations in other works. Hence our results provide insights into the effects of such optimization decisions across the layers. The results were consistent with results obtained by previous work. We were able to provide insights from

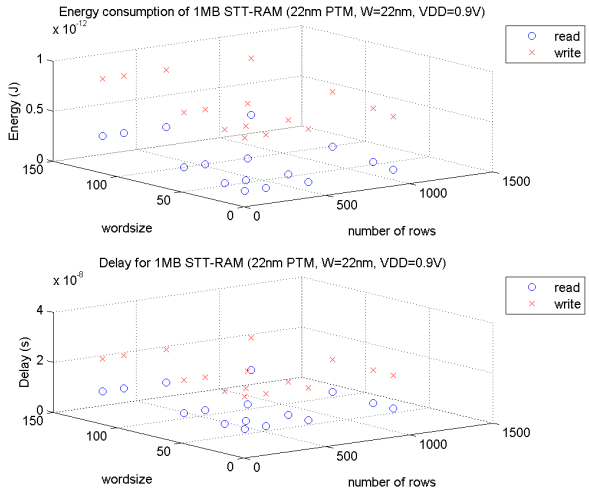

Figure 5: Array-level parameter effects on energy and delay. Word size is the number of bits being read or written as a memory word

different perspectives using the same set of tools, whereas other works focused on insights from one perspective (varying circuit level parameters or array-level parameters). In general, the write energy is expected to be greater than the read energy, which we observed. Also, the write delay is expected to be larger than the read delay, though writes could be faster for higher currents as seen in the process technology results where larger transistors are used. Some of our results were limited by the models used to derive the write-time for the bit cell. Since this value cannot yet be obtained in simulation, we used a modified version of model provided by [3] and [9]. This model only accounts for write times greater than 10ns and hence for write currents where write times are less than 10ns, some of the numbers may be off. Our main aim, however was to show that decisions in each layer could be reflected at the array level using the same set of tools.

7. FUTURE WORK

The main work to be done is to refine our STT-RAM models. We used a combination of models from different works and tried as best as we could to modify them to make them consistent; however, more work needs to be done to improve the consistency of our models, since most of the works used an MTJ model different from the one we used in our investigations. Also, we experienced problems with the MTJ model since it was originally developed in SPICE and we had to port it to Spectre. Being able to obtain the write time of the MTJ from simulation would help our work considerably, since the analytical models do not provide a way to determine which regime of operation the MTJ is in based on the current (they only work based on intended write pulse width). The MTJ has three different switching regimes based on the intended write pulse width (see [3] for a discussion on this property of the MTJ). The second part of our future work is to integrate the STT-RAM extensions with the original versions of TASE and ViPro. We developed our extensions outside of the original tools since these tools are still a work-in-progress and bugs and unresolved issues may have gotten in the way of our development. Other things we may consider is other outputs like sensitivity numbers which may aid designers in sensitivity-based optimization.

8. CONCLUSION

In this paper, we argued that a cross-layer perspective is necessary for early design space exploration in STT-RAM design. We showed how two tools that provide such a perspective for SRAM could be extended to provide similar insights for STT-RAM. We described how STT-RAM models are introduced into the tools and presented our preliminary results based on varying parameters on the process, circuit, and architecture levels. Our results were consistent with similar results in the literature, however there is still a need to refine our models and build on this work to improve the accuracy of our results.

9. REFERENCES

[1] T. Blalock and R. Jaeger. A high-speed clamped bit-line current-mode sense amplifier. *Solid-State Circuits, IEEE Journal of*, 26(4):542 –548, apr 1991.

[2] S. Chatterjee, M. Rasquinha, S. Yalamanchili, and S. Mukhopadhyay. A scalable design methodology for energy minimization of sttram: A circuit and architecture perspective. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 19(5):809 –817, may 2011.

[3] A. Jog, A. K. Mishra, C. Xu, Y. Xie, N.Vijaykrishnan, R. Iyer, and C. R. Das. Cache revive: Architecting volatile STT-RAM caches for enhanced performance in CMPs. Technical Report CSE 11-010, Computer Science and Engineering Department, Pennsylvania State University, 2011.

[4] S. Nalam, M. Bhargava, K. Mai, and B. H. Calhoun. Virtual prototyper (vipro): An early design space exploration and optimization tool for sram designers. In *Design Automation Conference (DAC), 2010 47th ACM/IEEE*, pages 138 –143, june 2010.

[5] S. Nalam, M. Bhargava, K. Ringgenberg, K. Mai, and B. Calhoun. A technology-agnostic simulation environment (TASE) for iterative custom IC design across processes. In *Computer Design, 2009. ICCD 2009. IEEE International Conference on*, pages 523 –528, oct. 2009.

[6] A. Nigam, K. Munira, A. Ghosh, S. Wolf, E. Chen, and M. Stan. Self consistent parameterized physical MTJ compact model for STT-RAM. In *Semiconductor Conference (CAS), 2010 International*, volume 02, pages 423 –426, oct. 2010.

[7] A. Nigam, C. Smullen, V. Mohan, E. Chen, S. Gurumurthi, and M. Stan. Delivering on the promise of universal memory for spin-transfer torque RAM (STT-RAM). In *Low Power Electronics and Design (ISLPED), 2011 International Symposium on*, pages 121 –126, aug. 2011.

[8] C. Smullen, V. Mohan, A. Nigam, S. Gurumurthi, and M. Stan. Relaxing non-volatility for fast and energy-efficient STT-RAM caches. In *High Performance Computer Architecture (HPCA), 2011 IEEE 17th International Symposium on*, pages 50 –61, feb. 2011.

[9] C. Smullen, A. Nigam, S. Gurumurthi, and M. Stan. The STeTSiMS STT-RAM simulation and modeling system. In *Computer-Aided Design (ICCAD), 2011 IEEE International Conference on*, Nov. 2011.

[10] M. Turi and J. Delgado-Frias. Reducing power in memory decoders by means of selective precharge schemes. In *Circuits and Systems, 2007. MWSCAS 2007. 50th Midwest Symposium on*, pages 956 –959, aug. 2007.

[11] S. Wolf, J. Lu, M. Stan, E. Chen, and D. Treger. The promise of nanomagnetics and spintronics for future logic and universal memory. *Proceedings of the IEEE*, 98(12):2155 –2168, dec. 2010.